on

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/693,822	10/23/2003	Leonardo E. Blanco	4090	1433

7590 06/28/2006

Law Offices of Albert S. Michalik, PLLC
Suite 193
704-228th Avenue NE
Sammamish, WA 98074

EXAMINER	
HAJNIK, DANIEL F	
ART UNIT	PAPER NUMBER
2628	

DATE MAILED: 06/28/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/693,822	**Applicant(s)** BLANCO ET AL.	
	Examiner Daniel F. Hajnik	**Art Unit** 2628	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *4/5/2006*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-36* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 October 2003* is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

Response to Amendment

1. Claims 1, 18, and 29 have been amended.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Milne (US Patent 5553222, herein referred to as "Milne") in view of Grinstein et al. (US Patent 6714201, herein referred to as "Grinstein").

As per claim 1, Milne teaches the claimed:

> 1. In a computing environment, a system comprising:
>
> a first component that receives clock data from a program;

in figure 9 where Player 900 is the first component which receives clock data (910) from a program. Further, clock 500 in figure 5 can also be a first component because it receives clock data to tick every 1,000,000 units.

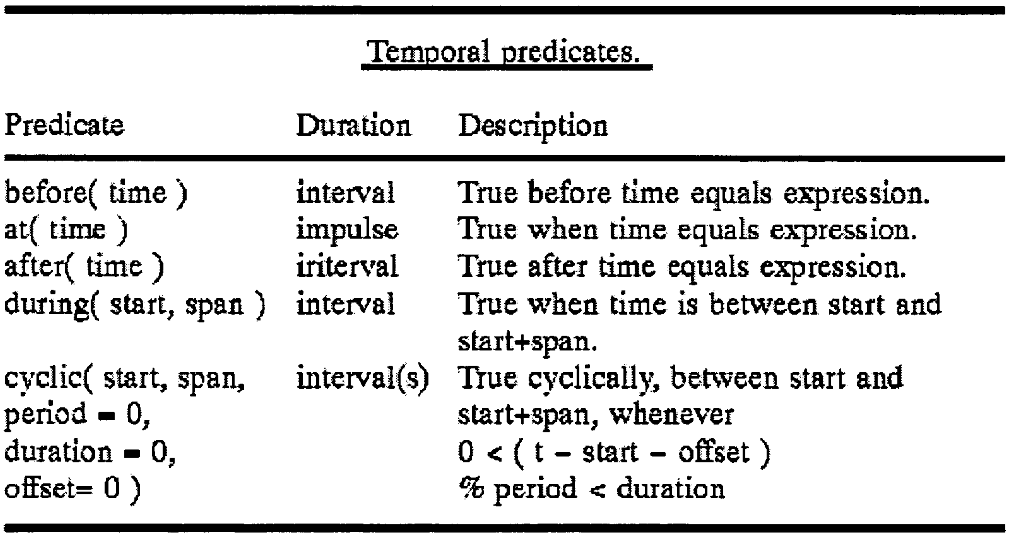
Milne teaches the claimed "a second component" by teaching of a clock 510 in figure 5 (a second component).

Milne does not explicitly teach the claimed:

> an interval generation mechanism that computes interval data based on the clock data, wherein the interval data corresponds to a relative determination of time between a first event and a second event; and

Grinstein teaches the claimed limitation by teaching of in table 9 (col 25):

TABLE 9



Temporal predicates.

Predicate	Duration	Description
before(time)	interval	True before time equals expression.
at(time)	impulse	True when time equals expression.
after(time)	iriterval	True after time equals expression.
during(start, span)	interval	True when time is between start and start+span.
cyclic(start, span, period - 0, duration - 0, offset= 0)	interval(s)	True cyclically, between start and start+span, whenever 0 < (t - start - offset) % period < duration

where an interval is defined in terms 'during (start, span)'. In other words, the table shows that the duration of interval is defined as the difference of time between when the start and start+span events trigger. Further, an interval can be defined as cyclic (start, span, period, duration, offset) in order to associated a specific % period of a given duration (relative determination of time between a first event and second event). Grinstein further teaches the claimed limitation by teaching of:

> The dispatch package 140 is a group of related classes which the RTE uses to **update motions**, **schedule events**, and **represent time**. These include the Clock class 142, the Dispatcher class 146, and the ScheduledItem class 148. **The clock class is used to define a computational object which represents the passage of time**. The dispatcher class determines the order in which Behavior, ScheduledItem, and Motion class instances are updated. The **ScheduledItem** class, **defines items which require updating at a given time**
> (col 10, lines 57 – 65, some text bolded for clarity)

Here, the clock class and ScheduledItem class allow for the defining of intervals of time and for interactive events to be scheduled.

Thus, through the combination of Milne and Grinstein, all the claimed limitations are taught by the prior art. It would have been obvious to one of ordinary skill in the art at the time of invention to combine Milne and Grinstein. Grinstein teaches one advantage of the combination by teaching of interactive animation motion control specifically by teaching of:

> FIGS. 43 and 44 illustrates the **change-simulation-clock-speed** window 532 that will be displayed if the user clicks the clock button 528. The window 532 includes a slider control 532A which **enables a user to very the speed of the simulation clock** for the entire animation shown in the scene view window 503.
> (col 58, lines 40-46)

where Milne would benefit from such added functionality.

As per claim 18, the reasons and rationale for the rejection of claim 1 are incorporated herein in regards to the claimed "generating interval data".

Milne teaches the claimed "receiving clock data" in figure 3 where a clock receives clock data from the system timer.

Milne teaches the claimed "causing output to be produced" by showing in figure 14 an audio player and video player (also see figure 28).

As per claim 2, Milne teaches the claimed:

> 2. The system of claim 1 wherein the output corresponds to a progress of an animation having an animated characteristic.

in figure 28 where the figure shows images (animated characteristic).

As per claim 3 Milne does not explicitly teach the claimed limitations.

Grinstein teaches the claimed:

> 3. The system of claim 1 wherein the second component determines the output by interpolating a current progress value for the animated characteristic.

By teaching of:

> Define paths with key frames & scripting **interpolating** function (table 3, col 15)

where such interpolating can be used with animation and key frames to achieve the claimed limitations.

This is similar to the interpolating and progress value described in the submitted application which teaches of:

> **a progress value** based on the interval data for that object and the current time to an low-level animation function subsystem 2620.sub.L that determines a current value for the varying property of an animated object. **For example**, for any given frame, the low-level computation engine 2514 **interpolates the location of the animated object** based on the interval data and the current time.
> (specification, pg. 60, lines 11 – 18)

In Grinstein progress is also determined over time through interpolation, such as changing the location of an animated object between keyframes. Grinstein further teaches the claimed progress value by teaching of

> The Period is how long a particular motion cycle lasts, and the Active Cycle controls **how long its active within the Period**. For example, a **Spin motion** may have a Period of 10 sec. and an **Active Cycle of 20%**. The object will spin (become active) for 2 sec. and remain still for 8 seconds.
> (col 51, lines 15 – 20)

where 20% indicates animation progress based upon interval data and current time data. It would have been obvious to use this claimed feature as taught by Grinstein. One advantage to utilizing such interpolation techniques is that it saves animation data and reduces animation file size.

As per claim 4, Milne teaches the claimed limitation of:

> 4. The system of claim 1 wherein the second component computes the output at a fast operating rate relative to an operating rate of the first component.

in figure 5 where clock B (second component) is going faster (an thus faster outputting) relative to clock A (first component).

As per claim 5, Milne does not teach the claimed limitations.

Grinstein teaches the claimed limitations of:

> 5. The system of claim 1 further comprising an event list generation mechanism, wherein the interval generation mechanism computes the interval data from an event list provided by the event list generation mechanism, the event list based on the clock data.

By teaching of:

> FIGS. 56A through 56C represent just some of the instructions in an event loop 640 of the Mojo editor. The **event loop** defines the response which the program will make to each of **a plurality of different events**, including events corresponding to user input, events generated by the Mojo program itself, events generated by the OpenMotion API with which the Mojo motion editor is used, and events generated by the operating system (col 62, lines 13 – 20)

and further in table 1, where the table is indicates 'Events' and correspondingly 'Actions occur periodically and at scheduled times'. Here, the event loop is maintaining a plurality of different events (event list). Lastly, Grinstein teaches of computing interval data based upon the event list by teaching of:

> The dispatch package 140 is a group of related classes which the RTE uses to update motions, **schedule events**, and **represent time**.... **The clock class is used to define a computational object which represents the passage of time**. ... The **ScheduledItem class**, **defines items which require updating at a given time**
> (col 10, lines 57 – 65)

It would have been obvious to use this claimed feature as taught by Grinstein. One advantage to utilizing such events loops is more interactivity and interesting scenes (i.e. see table 1 of Grinstein).

As per claim 6, Milne does not teach the claimed limitations.

Grinstein teaches the claimed limitations of:

> 6. The system of claim 5 wherein the first component receives an interactive event, and wherein the event list generation mechanism further adds the interactive event into the event list.

By teaching of:

> FIGS. 56A through 56C represent just some of the instructions in an event loop 640 of the Mojo editor. The **event loop** defines the response which the program will make to each of **a plurality of different events**, including **events corresponding to user input**, events generated by the Mojo program itself, events generated by the OpenMotion API with which the Mojo motion editor is used, and events generated by the operating system (col 62, lines 13 – 20)

where an event corresponding to user input is interactive. Further, this event would need a way of being added otherwise the system would not consider it. Grinstein teaches one advantage of the event list by teaching of in table 1 (col 14) of constantly changing and appealing activities in screen savers.

As per claim 7, Milne does not teach the claimed limitations.

Grinstein teaches the claimed limitations of:

> 7. The system of claim 6 wherein the event list generation mechanism further adds at least one implicit event into the event list.

By teaching of:

> The **Temporal tab overrides the continuous action of a motion** if the Active button in checked. Examples of temporal tabs are shown in FIGS. 11, 14, 17 and 21. The **Start Time and Stop Time** are tied to the **sim time**. The Period is how long a particular motion cycle lasts, and the Active Cycle controls how long its active within the Period.
> (col 51, lines 12 – 18)

where overriding can include placing an implicitly end event before the sim time is changed to begin again. It would have been obvious to use this claimed feature as taught by Grinstein. One advantage to utilizing such interactivity through start and stop overrides is more interesting controls for the user to entertain themselves with.

As per claim 8, Milne does not teach the claimed limitations.

Grinstein teaches the claimed limitations of:

> 8. The system of claim 6 wherein the event list generation mechanism marks at least one event in the event list as unused, the interval generation mechanism not using an unused event in computing the interval data.

By teaching of:

> The Period is how long a particular motion cycle lasts, and the Active Cycle controls **how long its active within the Period**. For example, a **Spin motion** may have a Period of 10 sec. and an Active Cycle of 20%. The object will spin (**become active**) for 2 sec. and **remain still** for 8 seconds.
> (col 51, lines 15 – 20)

Here, in regards to the Spin motion event, the unused event results in the object remain still (in particular no spinning for 8 seconds). More specifically, Grinstein teaches of:

> FIGS. 56A through 56C represent just some of the instructions in **an event loop 640 of the Mojo editor**. The **event loop** defines the response which the program will make to each of **a plurality of different events**, including events corresponding to user input, **events generated by the Mojo program itself**, events generated by the OpenMotion API with which the Mojo motion editor is used, and events generated by the operating system
> (col 62, lines 13 – 20)

Here, the event list utilizes events generated by the Mojo editor. If the reference indicates that the Mojo editor wants there to be a spinning motion for 2 second, then the Mojo editor wants this spinning to stop and remain still. Subsequently, the Mojo editor will have to indicate this stopping (or not using the spinning event) to the event loop (i.e. through marking the event as unused to the event loop) since the event loop defines the response of the program to the eventual output screen.

It would have been obvious to use this claimed feature as taught by Grinstein. One advantage to utilizing such unused events is to better stop and start animation motions such as spinning.

As per claim 9, Milne teaches the claimed:

> 9. The system of claim 1 wherein the clock data comprises property information corresponding to a begin time value and a duration.

By teaching of:

> A time-based media sequence ... MIDI ... It **starts at time 0** and **has a duration**
> (col 8, lines 63-67)

where the MIDI sequence is associated with a clock and thus shares these properties with the clock (also see col 9, lines 7-10 and 20-23).

As per claim 10, Milne teaches the claimed:

> 10. The system of claim 1 wherein the clock data comprises property information corresponding to a repeat count.

in figure 5 where clock A is shown to have a repeat count of 2 where the repeat count indicates that clock B ticks at least twice as often as clock A. For example, clock A waits for clock B to be repeated twice before adding a unit of time to its count.

As per claim 11, Milne teaches the claimed:

> 11. The system of claim 1 wherein the clock data comprises property information corresponding to a reverse instruction.

by stating "Clocks can travel backwards in time" (col 7, lines 28).

As per claims 12 and 13, Milne teaches the claimed:

> 12. The system of claim 1 wherein the clock data comprises property information corresponding to acceleration data.
>
> 13. The system of claim 1 wherein the clock data comprises property information corresponding to deceleration data.

By teaching of basing a moving playback position (which is the equivalent of a play head on a tape recorder) according to a clock rate (col 9, lines 12-16). Milne teaches of slowing down and speeding up a clock such as a master clock (col 9, lines 30-33) where this slowing down and speeding up would have to have an associated de-acceleration or acceleration. Milne teaches of manipulating more than one clock speed as well through multiple audio and video sequences (col 9, lines 51-60).

As per claim 14, Milne teaches the claimed:

> 14. The system of claim 1 wherein the clock data comprises property information corresponding to a seek instruction.

By teaching of:

> A non-driven time source knows how to find its current time, and it has a member function, GetNextTime(), that returns the next time that an alarm or delay should be fired
> (col 12, lines 57-60)

where this process of finding the next time an alarm or delay should be fired is a seek instruction because it is seeking out the next time an associated event should fire.

As per claim 15, Milne teaches the claimed:

> 15. The system of claim 1 wherein the clock data comprises property information corresponding to speed data.

By teaching of a clock rate (speed data), specifically, by stating

> **a is a floating point value** that **determines the rate** of the slave clock's current time **relative** to the master clock's current time)
> (col 8, lines 25-27).

As per claim 16, Milne teaches the claimed:

> 16. The system of claim 1 wherein the clock data comprises property information corresponding to function data.

By teaching of

> A software clock, an illustration of which appears in FIG. 2, is an object that **performs the following functions** ... **Member functions** are provided for setting and getting the current time
> (col 7, lines 24-25 and lines 29-30).

As per claim 17, Milne teaches the claimed:

> 17. The system of claim 1 wherein the first component and second component execute on different threads.

By teaching of associating different clocks (and thus their associated players which are components) with a unique thread by teaching of blocking/unblocking threads. Milne states:

> A clock can block **a thread** until a certain time, called the delay time, is reached. If the clock is going forward, **the thread is unblocked** when the clock's current time is equal to or greater than the delay time
> (col 7, lines 35-39).

Here, the reference teaches of animation or audio sequence executing on a different threads, i.e. where in figure 10 sequence Player B is the second component and sequence Player A is the first component. Figure 10 further shows that these components have associated clocks.

As per claims 19-20, Milne does not teach the claimed limitations. Grinstein teaches the claimed:

> 19. The method of claim 18 wherein causing output to be produced based on the current time data and the interval data comprises, determining an interval, and determining a progress value within that interval.
>
> 20. The method of claim 19 further comprising, causing an animation property value to be determined based on the progress value, such that the animation property value varies as the current time varies

By teaching of:

> The Period is how long a particular motion cycle lasts, and the Active Cycle controls how long its active within the Period. For example, **a Spin motion may have a Period of 10 sec. and an Active Cycle of 20%**. The object will spin (become active) **for 2 sec**. and remain still **for 8 seconds.** (col 51, lines 15 – 20)

Here, Grinstein determines a progress value by teaching of determining an active state for 20% of the time. Here, an animation property value varies (active cycle control) as the current time varies.

As per claims 21-24, the claimed limitations are similar to those of claims 5-8, respectively, in terms of functionality and thus are subject to the same reasons for rejection.

As per claim 25, Milne does not explicitly teach the claimed limitation.

Grinstein teaches the claimed:

> 25. The method of claim 18 wherein causing output to be produced based on current time data and the interval data comprises constructing a frame at a rate that corresponds to a frame refresh rate of a graphics subsystem.

By teaching of:

> If om::**frame( ) is not used**, the motions will not be affected, but the **display will lag behind the time** for which OpenMotion calculates positions by the **graphic** and **application update latency**
> (col 16, line 66 - col 17, line 3)

As per claim 26, Milne teaches in figure 1 the claimed "computer-readable medium" in piece 14, labeled 'RAM" and teaches the claimed "computer-executable instructions" in piece 10, labeled 'CPU'.

3. Claims 27 and 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Milne in view of Cheng et al. (US Patent 6731314, herein referred to as "Cheng").

As per claim 27, Milne teaches the claimed:

> a first field having data indicative of begin time;
>
> a second field having data indicative of an initial progress value;
>
> a third field having data indicative of an end time;
>
> a fourth field having data indicative of an final progress value; and

By teaching of

> A time-based media sequence is an abstract base class that can be used **to represent a clip of audio, video, animation, or Musical Instrument Digital Interface (MIDI) data**, or any other data that **varies over time**. It starts at time 0 and **has a duration represented by a time object**. FIG. 7

> is an example of a time based sequence that is **three seconds in duration**.
> (col 8, line 63 – col 9, line1)

Further, figure 8 shows the starting time (first field), the play position (second field), the ending time (third field), and fourth field would be contained within the shown 'Time-Based Media Sequence'.

Milne does not teach the claimed:

> Wherein a current time between the begin time and the end time is used to interpolate a progress value between the initial progress value and the final progress value

Cheng teaches the claimed limitation by teaching of:

> A technique for efficiently authoring and storing information about an object with **respect to time** ... This technique **only stores the information** about an object's properties **at specific times**, known as key frames. When the animation is played back, **times between key frames are interpolated** to give the appearance of smooth animation
> (col 23, line 61 – col 24, line 2).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine Cheng with Milne. Cheng provides one advantage by teaching of achieving smooth animation by interpolating (col 23, line 67 – col 24, line 2).

As per claim 28, Milne teaches the claimed:

> 28. The data structure of claim 27 further comprising a fifth field indicative of an iteration.

By showing an iteration in figure 2 where the clock has an iteration of relating mathematically to picoseconds (also see col 7, line 27-29).

4. Claims 29 and 32-36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Anderson (US Patent 5986675, herein referred to as "Anderson") in view of Grinstein.

As per claim 29, Anderson teaches the claimed:

> 29. In a computing environment, a method comprising:
>
> generating an event list based on scheduled events and at least one interactive event;

in figure 19 (also see col 28, lines 5-12).

Anderson teaches the claimed:

> processing data associated with the current interval to produce an output based on the time value

in figure 18 where arrows are used to show a current interval in a given interval sequence. Further, Anderson teaches of a 3D animated movie (output) (see abstract).

Anderson does not explicitly teach the claimed:

> computing an interval list based on the event list;
>
> determining a current interval in the interval list based on a time value, wherein the interval data corresponds to a relative determination of time between a first event and a second event

Grinstein teaches claimed:

> computing an interval list based on the event list;

By teaching of:

> The dispatch package 140 is a group of related classes which the RTE uses to **update motions**, **schedule events**, and **represent time**. ... The ScheduledItem class, **defines items which require updating at a given time**
> (col 10, lines 57 – 65, event list)
>
> The Period is how long a particular motion cycle lasts, and the Active Cycle controls **how long its active within the Period**. For example, a Spin motion **may have a Period of 10 sec**. and an Active Cycle of 20%. The object **will spin** (become active) for **2 sec**. and **remain still for 8 seconds**.
> (col 51, lines 15 – 20, intervals)

Here, an interval list can be based upon schedule events (event list). The interval list has an interval for 2 seconds and then for 8 seconds.

Grinstein teaches claimed:

> determining a current interval in the interval list based on a time value, wherein the interval data corresponds to a relative determination of time between a first event and a second event

For the same reasons as stated in the rejection of claim 1.

It would have been obvious to one of ordinary skill in the art to combine Anderson and Grinstein. Grinstein teaches one advantage of the combination by teaching of interactive animation motion control specifically by teaching of:

> FIGS. 43 and 44 illustrates the **change-simulation-clock-speed** window 532 that will be displayed if the user clicks the clock button 528. The window 532 includes a slider control 532A which **enables a user to very the speed of the simulation clock** for the entire animation shown in the scene view window 503.
> (col 58, lines 40-46)

where Anderson would benefit from such added functionality.

As per claim 32, Anderson does not explicitly teach the claimed:

> 32. The method of claim 29 wherein generating the event list comprises determining how a current state is modified by an event, and when the event modifies the current state to another state that cannot be directly transitioned to by that event, inserting at least one implicit event to transition the current state to the other state via an intermediate state.

Grinstein teaches claimed limitations for the same reasons as the claimed "adds at least one implicit event into the event list" of claim 7. Further, Grinstein teaches of simulation clock controls in table 4 (col17) where an event (i.e. pause or resume) can modify a current event. Further, table 9, indicates that the intervals need a corresponding start and stop (start + span) events in order to be timed. Such events can require inserting implicit events (i.e. start or stop) when an event has been paused or resumed.

As per claims 33 and 34, Anderson does not explicitly teach the claimed:

> 33. The method of claim 32 wherein the current state is paused, the event corresponds to a begin event, and wherein inserting at least one implicit event comprises inserting an end event before the begin event such that the state transitions to the intermediate state comprising the inactive state transitions before transitioning to the other state corresponding to the active state.
>
> 34. The method of claim 29 wherein generating the event list comprises completing an iteration by inserting an implicit end event into the event list and starting a next iteration by inserting an implicit begin event into the event list.

However, Anderson in figure 17 shows a 'Record new path' in step 1730, and an 'End record' in step 1720 which are associated with adding events into the event list and associated with editing the event list. Given that the system would need to track when the recording starts and ends, it would have been obvious to include implicit events to mark these actions. One advantage of these implicit events is to better determine a time order of the user's interactive actions editing events (col 28, lines 5-12 and 15-16).

As per claim 35, Anderson does not explicitly teach the claimed:

> 35. The method of claim 29 wherein generating the event list comprises, marking an event as unused when the event will not change a state of operation.

Grinstein teaches the claimed limitation for the same reasons as stated in the rejection of claim 7.

As per claim 36, Anderson teaches the claimed:

> 36. (original) A computer-readable medium having computer-executable instructions for performing the method of claim 29.

In figure 1, piece 21.

5. Claims 30 and 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Anderson in view of Grinstein in further view of Milne.

As per claim 30, Anderson does not explicitly teach the claimed:

> 30. The method of claim 29 wherein generating the event list comprises receiving clock properties.

Milne teaches the claimed limitation in figure 32, step 3220, labeled 'Get Clock Time'.

It would have been obvious to one of ordinary skill in the art to combine Anderson, Grinstein, and Milne. Milne teaches one advantage of the combination by teaching of synchronized multimedia events using clock properties (col 1, line 64 – col 2, line 3).

As per claim 31, Anderson does not explicitly teach the claimed:

> 31. The method of claim 30 further comprising, generating another event list based on a relationship between the clock properties and properties of another clock.

Milne teaches of a dependent clock receiving properties from another clock (i.e. an independent clock) in figure 15. If the event list was based upon a dependent clock the event list would also be dependent upon the independent clock.

Response to Arguments

6. Applicant's arguments filed have been fully considered but they are not persuasive.

Applicants argues the prior art rejection of independent claim 1. However, these arguments are based upon recently amended claim language and are now moot in view of new grounds of rejection to address these amendments.

Applicant further argues:

> For example, claim 9 recites system of claim 1 wherein the clock data comprises property information corresponding to a begin time value and a duration. Milne may teach that clock data describes a begin time value regarding an event. However, Milne is completely silent with respect to determining anything about the duration of an event with respect to another event, such that other events may be relative to the duration of

> the initial event. For at least this additional reason, applicants submit that claim 9 is allowable over the prior art of record.
> (middle of pg. 15 of remarks)

Milne teaches the claimed limitations of:

> 9. The system of claim 1 wherein the clock data comprises property information corresponding to a begin time value and a duration.

By stating "A time-based media sequence ... MIDI ... It starts at time 0 and has a duration" (col 8, lines 63-67) where the MIDI sequence is associated with a clock and thus shares these properties with the clock (col 9, lines 7-10 and 20-23). This claim has no specific mention to duration of event in respect to another event, and thus the prior art rejection is proper. It claims a begin time and duration, which are taught by Milne.

Applicant further argues the rejection of claim 18. However, these arguments are based upon recently added claim language and are further moot in view of new grounds of rejection.

Applicant further argues:

> Applicants submit that the Office action has failed to establish a prima facie case for obviousness. Not all of the recitations of claim 27 have be shown to be taught or suggested by the prior art of record. Specifically, as discussed above, interval data is a concept that is neither taught nor suggested by the prior art of record. In claim 27, interval data is expressed as an interpolated value between begin time and end time while taking into consideration current time and current progress.
> (page 18 of remarks)

The prior art rejection of claim 27 is proper. Claim 27 does not explicit recite a claimed "interval".

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., interval) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Specifically, Cheng teaches the following limitation of claim 27:

> Wherein a current time between the begin time and the end time is used to interpolate a progress value between the initial progress value and the final progress value

By teaching of:

> A technique for efficiently authoring and storing information about an object with **respect to time** ... This technique **only stores the information** about an object's properties **at specific times**, known as key frames. When the animation is played back, **times between key frames are interpolated** to give the appearance of smooth animation (col 23, line 61 – col 24, line 2).

Here, the claimed limitations are taught where a current time between the begin time (a previous keyframe) and end time (the next keyframe) is used to interpolate the time between keyframes.

Lastly, applicants argues the prior art rejection of independent claim 29. However, these arguments are based upon recently amended claim language and are now moot in view of new grounds of rejection to address these amendments.

Conclusion

7. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Daniel F. Hajnik whose telephone number is (571) 272-7642. The examiner can normally be reached on Mon-Fri (8:30A-5:00P).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ulka J. Chauhan can be reached on (571) 272-7782. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DFH

Ulka Chauhan

ULKA CHAUHAN
SUPERVISORY PATENT EXAMINER